

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/233779415>

The decision automata for manoeuvre planning and control in simulation aided system for operational training

Conference Paper · September 2007

CITATIONS

2

READS

41

1 author:



[Zbigniew Tarapata](#)

Military University of Technology

105 PUBLICATIONS 550 CITATIONS

SEE PROFILE

THE DECISION AUTOMATA FOR MANOEUVRE PLANNING AND CONTROL IN SIMULATION AIDED SYSTEM FOR OPERATIONAL TRAINING

Zbigniew Tarapata
Military University of Technology
Cybernetics Faculty
Kaliskiego 2
00-908 Warsaw, Poland
E-mail: zbigniew.tarapata@wat.edu.pl

ABSTRACT

The idea and model of command and control process applied for the decision automata for march on the battalion level are considered. The automata executes two main processes: decision planning process and direct march control. The decision planning process relating to the automata contains: march organization (units order in march column, count and place of stops), determining paths for units, determining detailed march schedule for each unit in the column. The direct march control process contains such phases like command, reporting and reaction to fault situations during march simulation. The implementation process and experimentation approach are presented as well. Some results of march planning and realization (simulation) in real simulation aided system for operational training are presented and discussed.

INTRODUCTION

The automation of simulated battlefield is a domain of Computer Generated Forces (CGF) systems [12], [16] or semi-automated forces (SAF or SAFOR). CGF or SAF (SAFOR) are the technique which provides the simulated opponent using a computer system that generates and controls multiple simulation entities using software and possibly a human operator. In the case of Distributed Interactive Simulation (DIS) systems, the system is intended to provide a simulated battlefield which is used for training military personnel. One of the element of the CGF systems is movement planning and simulation of military objects. Any module for redeployment

modeling forces is part of such a software. For example in *ModSAF* (*Modular Semi-Automated Forces*) in module “SAFsim”, which simulates the entities, units, and environmental processes the route planning component is located [13]. In the paper [14] implementation of a *Tactical Mission Planner* for command and control of Computer Generated Forces in *ModSAF* is presented. In many of existing simulation systems there are different solutions regarding to this subject. In the *JTLS* system [24] terrain is represented using hexagons with size from 1km to 16km. In the *CBS* system [6] terrain is similarly represented but additionally vectoral-region approach is applied. In both of these systems there are manual and automatic methods for route planning (e.g. in the *CBS* controller sets intermediate points (coordinates) for route). In the work [3] authors describe a combined on-road/off-road planning system that was closely integrated with a geographic information system and a simulation system. Routes can be planned for either single columns or multiple columns. For multiple columns, the planner keeps track of the temporal location of each column and insures they will not occupy the same space at the same time. In the same paper the *Hierarchic Route Planner* as integrate part of *Predictive Intelligence Military Tactical Analysis System (PIMTAS)* is discussed. In the paper [9] authors presented an on-going efforts to develop a prototype for ground operations planning, the *Route Planning Uncertainty Manager (RPLUM)* tool kit. They are applying uncertainty management to terrain analysis and

route planning since this activity supports the Commander's scheme of maneuver from the highest command level down to the level of each combat vehicle in every subordinate command. They extend the PIMTAS [3] route planning software to accommodate results of reasoning about multiple categories of uncertainty. Authors of the paper [5] presented route planning in the *Close Combat Tactical Trainer (CCTT)*. Kreitzberg [11] has developed the *Tactical Movement Analyzer (TMA)*. The system uses a combination of digitized maps, satellite images, vehicle type and weather data to compute the traversal time across a grid cell. *TMA* can compute optimum paths that combine both on-road and off-road mobility, and with weather conditions used to modify the grid cost factors. The smallest grid size used is approximately 0.5 km. Author uses the concept of a signal propagating from the starting point and uses the traversal time at each cell in the array to determine the time at which the signal arrives at neighboring cells. In the paper [20] models and methods of movement planning and simulation in some simulation aided system for operational training on the corps-brigade level which has been realized in Cybernetics Faculty of Military University of Technology in Warsaw for Polish Army is described. A combined on-road/off-road planning system that is closely integrated with a geographic information system and a simulation system is considered. A dual model of the terrain, which is based at the digital map, is presented. The model is twofold: (1) as regular network of terrain squares, (2) as road-railroad network.

Regardless of kind of military actions military objects are moved according to some group (units arrangement). For example, each object being moved in group (e.g. during attack, during redeployment) must keep distances between each other of the group [22]. Therefore, it is important to recognize (during movement simulation) that objects inside units do not "keep" required distances (group pattern) and to determine a new movement schedule. All of presented above systems have not automatic procedures for synchronization movement of many units. The most often solution of this problem is one when

movement (and simulation, naturally) is stopped and commanders (trainees) make a new decision or system does not react such a situation. Therefore, in the paper [19] proposition of solution the problem of synchronization movement of many units is shown. Some models of synchronous movement and the idea of module for movement synchronization are presented.

In this paper the idea and model of command and control process applied for the decision automata for march on the battalion level are considered. These problems are very rarely discussed in the literature, however some ideas we can meet in [1], [2], [7], [8]. In the papers [1], [2] the idea and model of command and control process applied for the decision automata on the battalion level for two type of unit tasks: attack and defense, are presented. This paper complete description of the automata presented in [1], [2] for the third type of units task – march.

The automata for march executes two main processes: decision planning process and direct march control. The decision planning process relating to the automata contains determining: march organization (units order in march column, count and place of stops and rests), paths for units, detailed march schedule for each unit in the column. The direct march control process contains such phases like command, reporting and reaction to fault situations during march simulation. The automata is implemented in ADA language and it represents a commander of battalion level (the lowest level of trainees is brigade level). It is a component of distributed interactive simulation system SBOTSS "Zlocien" for CAX (Computer Assisted Exercises) [15].

THE TERRAIN MODEL

The terrain (environment) model S_0 which is used as battlefield model is based at the digital map in VPF format. The model is twofold: (1) as regular network Z_1 of terrain squares, (2) as road-railroad network Z_2 and it is defined as follows [20]:

$$S_0(t) = \langle Z_1(t), Z_2(t) \rangle \quad (1)$$

Regular grid of squares Z_1 divides terrain space into squares with the same size (200m×200m) and each square is homogeneous from the point of view of terrain characteristics (degree of slowing down velocity, ability to camouflage, degree of visibility, etc.). This size of square results from the fact that the nearest level of modeled units in SBOTSS “Zlocien” is platoon and 200m is approximately width of the platoon front during attack. The Z_1 model is used to plan off-roads (cross-country) movement e.g. during attack planning.

In the road-railroad network Z_2 we have crossroads as network nodes and section of the roads linking adjacent crossroads as network links (arcs, edges). This model is used to plan fast on-roads movement, e.g. during march (redeployment) planning.

These two models of terrain are integrated. This integration gives possibilities to plan movement inside both models. It is possible because each square of terrain contains information about fragments of roads inside this square. In the other side each fragment of the road has information about squares of terrain which cross. Hence, route for any object (unit) may consist of sections of roads and squares of terrain. It is possible to get off from road (if it is impassable) and start movement off-road (e.g. to omit impassable section of road) and next return to the road. In the other side we have move off-roads (e.g. during attack), get in section of the road (e.g. any bridge to go across the river) and next return to the off-road movement (on the second riverside). The characteristics of both terrain models depend on: time, terrain surface and vegetation, weather, the day and year season, opponent and own destructions (e.g. destruction of the bridge which is element of road-railroad network).

The model of regular network of terrain squares

The formal definition of the regular network of terrain squares Z_1 is following (see also Figure 1):

$$Z_1(t) = \langle G_1, \Psi_1(t) \rangle \quad (2)$$

where:

G_1 - Berge's graph defining structure of squares

network,

$$G_1 = \langle W_1, \Gamma_1 \rangle \quad (3)$$

W_1 - set of graph's nodes (terrain squares);

$\Gamma_1 : W_1 \rightarrow 2^{W_1}$ - function describing for each nodes of G set of adjacent nodes (maximal 8 adjacent nodes);

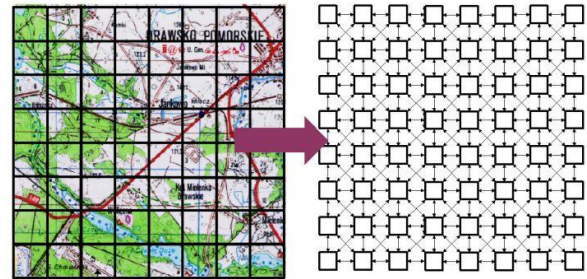


Figure 1. Regular grid of terrain squares (left-hand side) and its model Z_1 (right-hand side)

$\Psi_1(t) = \{\Psi_{1,0}(\cdot, t), \Psi_{1,1}(\cdot, t), \Psi_{1,2}(\cdot, t), \dots, \Psi_{1,L_{W_1}}(\cdot, t)\}$ - set of functions defined on the graph's nodes (depending on t).

One of the functions of $\Psi_1(t)$ is slowing down velocity function $FSDV(n, \dots)$, $n \in W_1$ which describes slowing down velocity (as real number from $[0,1]$) inside the n -th square of the terrain,

FSDV:

$$W_1 \times T \times K\_Veh \times K\_Meteo \times K\_YearS \times K\_DayS \rightarrow [0,1]$$

(4)

where:

T – time, K_Veh – set of kinds of vehicles, $K\_Veh =$

$\{\text{Veh\_Wheeled}, \text{Veh\_Wheeled-Caterpillar}, \text{Veh\_Caterpillar}\},$

K_Meteo – set of meteorological conditions, K_YearS – set of the year's seasons, K_DayS – set of the day's seasons;

We use function $FSDV$ to calculate crossing time between two squares of the terrain.

Other functions (as subset of $\Psi_1(t)$) described on the nodes (squares) of G_1 and essential from the point of view of trafficability and movement are following:

- geographical coordinates of node (centre of square):

$$FWSP : W_1 \rightarrow R^3 \quad (5)$$

- ability to camouflage in the square:

$$FCam : W_1 \times T \rightarrow [0,1] \quad (6)$$

- degree of terrain undulating in the square:

$$FUnd : W_1 \rightarrow [0,1] \quad (7)$$

- subset of node's set of Z_2 network which are located inside the square:

$$FW1OnW2 : W_1 \rightarrow 2^{W_2} \quad (8)$$

The model of road-railroad network

The formal definition of the road-railroad network Z_2 is following (see also Figure 2):

$$Z_2(t) = \langle G_2, \Psi_2(t), \zeta_2(t) \rangle \quad (9)$$

where:

G_2 - Berge's graph defining structure of road-railroad network,

$$G_2 = \langle W_2, U_2 \rangle \quad (10)$$

W_2 - set of graph's nodes (crossroads);

$U_2 \subset W_2 \times W_2$ - set of graph G_2 arcs (sections of roads);

$\Psi_2(t) = \{\Psi_{2,0}(\cdot, t), \Psi_{2,1}(\cdot, t), \dots, \Psi_{2,LW_2}(\cdot, t)\}$ -

- set of functions defined on the graph's G_2 nodes (depending on t).

$\zeta_2(t) = \{\zeta_{2,i}(\cdot, t)\}_{i=1, \overline{LG_2}}$ - set of functions defined on the graph's G_2 arcs (depending on t).

Functions (as subset of $\Psi_2(t)$ and $\zeta_2(t)$) which are essential from the point of view of trafficability and movement, described on the nodes and arcs of G_2 are following:

- slowing down velocity function $FSDV2(u, \dots)$, $u \in U_2$ which describes slowing down velocity (as real number from $[0,1]$) on the u -th arc (section of road) of the graph:

FSDV2:

$$U_2 \times T \times K\_Veh \times K\_Meteo \times K\_YearS \times K\_DayS \rightarrow [0,1] \quad (11)$$

- geographical coordinates of node (crossroad):

$$FWSP2 : W_2 \rightarrow R^3 \quad (12)$$

- node of Z_1 inside which node of Z_2 is located:

$$FW2OnW1 : W_2 \rightarrow W_1 \quad (13)$$

- subset of node's set of Z_1 network inside

which the arc is located:

$$FU2OnW1 : U_2 \rightarrow 2^{W_1} \quad (14)$$

- degree of terrain undulating on the arc:

$$FUnd : U_2 \rightarrow [0,1] \quad (15)$$

- length of the arc:

$$FLen : U_2 \rightarrow R^+ \quad (16)$$

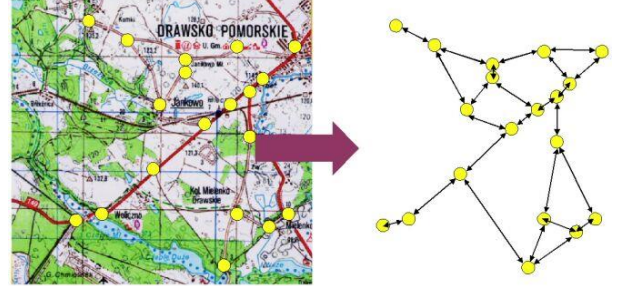


Figure 2. Road-railroad network (left-hand side) and its model Z_2 (right-hand side)

THE DECISION PLANNING PROCESS

The decision planning process relating to the automata contains determining such elements as: march organization (units order in march column, count and place of stops), paths for units, detailed march schedule for each unit in the column.

Algorithms which realize decision planning process described below are presented in the section "Automata implementation".

Decision process for march starts in the moment t , when the battalion id receives march order $SO(id, t)$ from superior (brigade) unit. Structure of the $SO(id, t)$ is as follows:

$$SO(id, t) = (t_0(id, t), t_s(id, t), MD(id, t)) \quad (17)$$

where:

$SO(id, t)$ - superior order for march for battalion id ;

$t_0(id, t)$ - readiness time for the unit id ;

$t_s(id, t)$ - starting time of the march for the unit id ;

$MD(id, t)$ - detailed description of march order.

Description of the $MD(id)$ (we omit t) is as follows:

$$MD(id) = \left\langle S(id), D(id), RP(id), IP(id) = (in_j(id), it_j(id))_{j=1, \overline{NIP}} \right\rangle \quad (18)$$

where:

$S(id), D(id)$ - source and destination areas for id , respectively;

$RP(id)$ – rest area for the id unit (after twenty-four-hours march), optional;

$IP(id)$ – vector of checkpoints for the id unit (march route must cross these points), $in_j(id)$ – the j -th checkpoint, $in_j(id) \in W_1 \cup W_2$, $in_1(id) = PS(id)$ is starting point of the march (in this point the head of march column is formed) and it is required, other checkpoints are optional, $it_j(id)$ – time of achieving the j -th checkpoint (optional);

NIP – number of checkpoints.

After receiving brigade commander order for march for the id unit (battalion) decision automata starts planning realization of this task. Taking into account $SO(id, t)$, for each unit id' (of company level and equivalent) directly subordinate to id the march order $MDS(id')$ is determined as follows:

$$MDS(id') = \left\langle \begin{array}{l} S(id'), D(id'), PS(id'), \\ PD(id'), RP(id'), \\ \mu(id', S(id'), D(id')) \end{array} \right\rangle \quad (19)$$

where:

$S(id'), D(id')$ - source and destination areas for id' , respectively, $S(id') \subset S(id)$, $D(id') \subset D(id)$;

$RP(id')$ – rest area for the id' unit (after twenty-four-hours march), $RP(id') \subset RP(id)$, optional parameter;

$PS(id')$ – starting point for the id' unit, the same for all $id' \in id$ and $PS(id') = in_1(id) \in W_1 \cup W_2$;

$PD(id')$ – ending point of the march for the id' unit, the same for all $id' \in id$ and $PD(id') \in W_1 \cup W_2$;

$\mu(id', S, D)$ - the route for the unit id' from the region $S(id') = S$ to the region $D(id') = D$,

$$\mu(id', S, D) = (w(id', m), v(id', m))_{m=1, LW(\mu(id', S, D))} \quad (20)$$

$w(id', m)$ - the m -th node on the path for id' , $w(id', m) \in W_1 \cup W_2$, $S, D \subset W_1 \cup W_2$ and $w(id', 1) \in S$, $w(id', LW(\mu(id', S, D))) \in D$;

$LW(\mu(id', S, D))$ – number of nodes (squares or crossroads) on the path $\mu(id', S, D)$ for id' unit;

$v(id', m)$ - velocity of the id' unit on the arc

starting in the m -th node.

Lets' note that path $\mu(id', S, D)$ may consist of sequences of nodes from $Z_1(t)$ and $Z_2(t)$ (when we accept descend from the road on the squares (if it is possible) and vice versa).

Detailed movement schedule for id' we define as follows:

$$H(id', t_0) = \langle S, D, \mu(id', S, D), T(id', S, D) \rangle \quad (21)$$

where:

t_0 – starting moment of schedule realization;

$T(id', S, D)$ - vector of moments of achieving nodes on the path,

$$T(id', S, D) = \langle t(id', m) \rangle_{m=1, LW(\mu(id', S, D))} \quad (22)$$

$t(id', m)$ - moment of achieving the m -th node of the path,

$$t(id', m) = t_0 + \sum_{k=1}^{m-1} \frac{L(w(id', k), w(id', k+1))}{v(id', k)} \quad (23)$$

$L(w(id', k), w(id', k+1))$ – geometrical distance between the k -th and the $(k+1)$ -st nodes of the path.

After determining $MDS(id')$ the order is sent by automata to subordinate unit.

Method of determining march route for unit is presented in the Figure 3.

Movement planning algorithms

To find paths for units modified shortest path algorithms (SPA): Dijkstra's, A*, geometric SPA are used. Modifications of mentioned algorithms deal with the following details:

- paths determining in different configurations: from point (region) to point (region), visiting selected points (regions), omitting selected points (regions, obstacles), inside or outside selected region, off-roads only, on-roads only, combined on- and off-roads and others;
- if we don't set region inside which we want to find the path then algorithm itself iterative determines rectangular region which is based on the line linking starting end ending points (nodes) of movement, to minimize computational time;
- if we want to find on-road path only, and there are not nodes of the road network

inside the intermediate squares then algorithm may optionally find crossroads (nodes of the road network) which are nearest to squares inside which the path must cross.

Detailed description of the movement planning algorithms used in SBOTTS “Zlocien” is presented in [20].

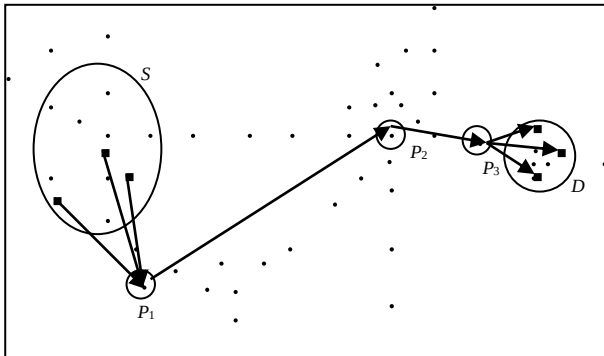


Figure 3. Example of march route (path) for three units (filled squares) from source area S to destination one D (dots represent crossroads from a digital map). We have three checkpoints: $P_1=PS$, P_2 , $P_3=PD$ (the path for all units must follow these points). The P_1 is starting point of the march (in this point the head of march column consisting of three units is formed), the P_3 is ending point of the march (in this point the march column is terminated in), the P_2 is intermediate point of the march. The path between P_1 and P_3 is common for all units, however each unit has own path from subarea of S to P_1 and from P_3 to subarea of D .

THE DIRECT MARCH CONTROL

The direct march control process contains such phases like: command, reporting and reaction to fault situations during march simulation.

Let's remember that automata replaces battalion commander and manages subordinate units (company or/and platoons and equivalent).

Identification and reaction to fault situations during march

The automata for march reacts following fault situations during march simulation:

1. current velocity of a subordinate unit differs from scheduled velocity;

Reaction:

- (a) if an unit is a head of column and it

does not move with planned velocity then increase velocity (in case of the delay) or decrease it (in case of the acceleration).

- (b) if an unit is not a head of column then adapt velocity to velocity of preceding unit.

A detailed description of movement synchronization is presented in the subsection “Velocity calculation”.

2. achieving of critical fuel level in one of subordinate unit;

Reaction:

reporting to automatic commander. Attempt to refuelling on the next stop or refuelling as soon as possible.

3. detection of an opponent unit;

Reaction:

if opponent forces are overwhelming (opponent combat potential is greater then some threshold value) and distance between own and opponent units is relatively small then unit is stopped, makes defence and reporting to commander. Otherwise, reporting to commander only.

4. detection of a minefield;

Reaction:

stopping and reporting to commander.

5. loss of capability to march realization (destruction of a part of the march route (e.g. bridge, river crossing), other cause of impassability);

Reaction:

- (a) if a part of the route is impassable because of destruction of a part of the march route then attempt to find detour. Reporting to commander.

- (b) if other cause of impassability then makes defence and reporting to commander.

6. contamination of a part of the march route or subordinate unit;

Reaction:

reporting to commander. If degree of contamination is low then running chemical defence and march continuation, otherwise try to exit from contaminated area.

Situations which require reporting to superior of battalion:

- (a) achieving checkpoints, stop area or rest area;

- (b) slowing down of velocity which causes delays;
- (c) contamination encountering;
- (d) minefield encountering;
- (e) achieving of fuel level with 75% and 50% of standard level;
- (f) loss capability of march realization (reporting cause of loss capability);
- (g) detection of opponent units.

Velocity calculation

We “see” the unit on the road twofold: (1) as occupying arcs (part of the roads) and nodes (crossroads) of the Z_2 network, (2) as sequence of squares of the Z_1 network by which the arc cross. In the (1) case we move the head and the tail of the column and we register arcs of the Z_2 on which the head and the tail are located with degrees of crossing these arcs. In the case (2) we locate the head and the tail of the column on small squares and we move “snake” of small squares (from the head to the tail). Movement of the unit on the road (deployed in the column) is done by determining sequence of nodes (crossroads) and arcs (part of the roads) of Z_2 network using algorithm presented in the section “The decision planning process” and next we realize movement from crossroad to crossroad.

The important problem during simulation is to set current velocity of the unit id . Procedure of velocity setting inside the k -th square consist of two cases: (a) when the unit id not fights in the k -th square; (b) when the unit id fights in the k -th square.

In the (a) case the current velocity $v_{cur}(id, k)$ of the unit id in the k -th square is calculated as follows:

$$v_{cur}(id, k) = \min\{v^{slowd}(id, k), v_{dec}(id, k)\} \quad (24)$$

where:

$v^{slowd}(id, k)$ - maximal velocity of the unit id in the k -th square taking into account topographical conditions,

$$v^{slowd}(id, k) = v^{max}(id) \cdot FOP(id, \bullet) \quad (25)$$

$v^{max}(id)$ - maximal possible velocity of the unit id resulting from technical parameters of the vehicles belonging to this unit,

$$v^{max}(id) = \min_{p \in ZVeh(id)} v^{tech}(p) \quad (26)$$

$ZVeh(id)$ – set of vehicles belonging to the id unit, $v^{tech}(p)$ – maximal velocity of the vehicle p (resulting from technical parameters), $FOP(id, \bullet)$ - slowing down velocity function for the id unit in the k -th square, $FOP(id, \bullet)$ is equal (4) or (11);

$v_{dec}(id, k)$ – velocity resulting from commander decision (equals $v(id, k)$ in (23)).

If the unit id is a head of column and it does not move with planned velocity $v_{dec}(id, k)$ then the velocity is increased (in case of the delay) or decreased (in case of the acceleration). If the unit id is not a head of column then velocity of the unit id is adapted to velocity of preceding unit.

In the (b) case the actual velocity $v_{akt}(id, k)$ of the unit id in the k -th square is calculated as follows:

$$v_{cur}(id, k) = \min \left\{ \begin{array}{l} f(v^{slowd}(id, \bullet), U_A, U_B, dist), \\ v_{dec}(id, k) \end{array} \right\} \quad (27)$$

where: $f(\bullet, \bullet, \bullet, \bullet)$ – function describing velocity in the square dependent on $v^{slowd}(id, \bullet)$, potentials of the unit id of the side A (U_A) and B (U_B) which fight, distance ($dist$) between fighting sides.

Some results of velocity calculations in real scenario for brigade march are presented in the Table 3.

Fuel consumption calculation

Fuel consumption $FC(id, veh, u)$ on the part u of a path for the type of vehicle veh belonging to the id unit is calculated as follows:

$$\begin{aligned} FC(id, veh, u) &= \\ &= FLen(u) \cdot FCC(u, veh) \cdot \frac{NFC(veh)}{100} \cdot N(id, veh) \end{aligned} \quad (28)$$

where:

$FLen(u)$ describes length of the u part of a path, $FCC(u, veh)$ – fuel consumption coefficient for the part u of a path and for vehicle type veh , $NFC(veh)$ – normative average fuel consumption for the veh type of vehicle (per 100km), $N(id, veh)$ – count of vehicles of type veh in the id unit.

Fuel consumption coefficient FCC is calculated as follows [4]:

$FCC(u, veh) = (1.0 + MTC(veh)) * (1.0 + UC(u))$;
where:

$MTC(veh)$ describes mechanical-tactical coefficient and $UC(u)$ - utilization coefficient, $veh \in K\_Veh$.

Calculation of $UC(u)$ is described in Table 1 and $MTC(veh)$ in Table 2. Some results of fuel consumption in real scenario for brigade march are presented in the Figure 9.

Table 1. Calculation of value of UC function (Utilization Coefficient)

Function <i>Get_UC</i> ($u \in U$)
<i>Und</i> := <i>FUnd</i> (<i>u</i>);
if <i>Und</i> <= 0.25 then
<i>UC</i> := 0.1;
elseif <i>Und</i> > 0.25 and <i>Und</i> <= 0.75 then
<i>UC</i> := 0.2;
else
<i>UC</i> := 0.25;
end if;
return <i>UC</i> ;
end <i>Get_UC</i> .

Table 2. Calculation of value of MTC function (Mechanical-Tactical Coefficient)

Function <i>Get_MTC</i> ($veh \in K\_Veh$)
if <i>veh</i> = Veh_Wheeled then
<i>MTC</i> := 0.25;
elseif <i>veh</i> = Veh_Wheeled-Caterpillar then
<i>MTC</i> := 0.27;
else
<i>MTC</i> := 0.3;
end if;
return <i>MTC</i> ;
end <i>Get_MTC</i> .

AUTOMATA IMPLEMENTATION

The automata is implemented in Ada language and it represents a part of automatic commander of battalion level. The automata realises its task and put the tasks for subordinate units. Simulation objects and their methods are managed by dedicated simulation kernel (extension of Ada language). Object

methods are divided into two sets: (1) non-simulation methods – designed in order to set and get attributes values, specific calculations and database operations; (2) simulation methods – prepared in order to synchronous (“waitfor” methods) and asynchronous (“tell” methods) data sending.

As it was written, in the papers [1], [2] the decision automata for two type of unit tasks: attack and defense, are presented. The automata for march completes this solution for automatic commander on the battalion level. They are components of distributed interactive simulation system SBOTSS “Zlocien” for CAX (Computer Assisted Exercises) [15]. For the decision planning process following procedures are implemented and used for each unit $id' \in id$:

Units_Order_In_March_Column_Determ(*id'*)
Column_Length_Determ(*id'*)
Number_of_Stops_Determ(*id'*)
Place_Of_Stops_Determ(*id'*)
Ending_Point_PD_Determ(*id'*)
March_Schedule_Determ(*id'*)
 Paths_Determ(*id'*)
 Path_S_To_PS_Determ(*id'*)
 Common_Path_PS_To_PD(*id'*)
 Path_PD_To_D_Determ(*id'*)
 Detailed_Schedule_Determ(*id'*)

For the direct march control process following procedures are implemented and used for each unit $id' \in id$:

March_Simulation(*id'*)
 Simulate_Unit_Movement(*id'*)
 React_To_Fault_Situations(*id'*)
 Fuel_Consumption_Determ(*id'*)
 Adapt_March_Velocity(*id'*)
 Report_To_Commander(*id'*)

PRACTICAL EXAMPLE

In this section practical example of march planning and simulation is presented. In the Figure 4 initial tactical situation is shown. In our example 2 mechanized brigades (121BZ and 123BZ: each of brigades consist of 4 mechanized battalion x 4 mechanized companies) of the blue side receive order to march.

In the superior order (17):

- destination area for 121BZ and 123BZ is set about 30 km to the north of the northern edge of location area of the red side;
- distance from source area S to destination one D is equal about 110km;
- 5 checkpoints is set.

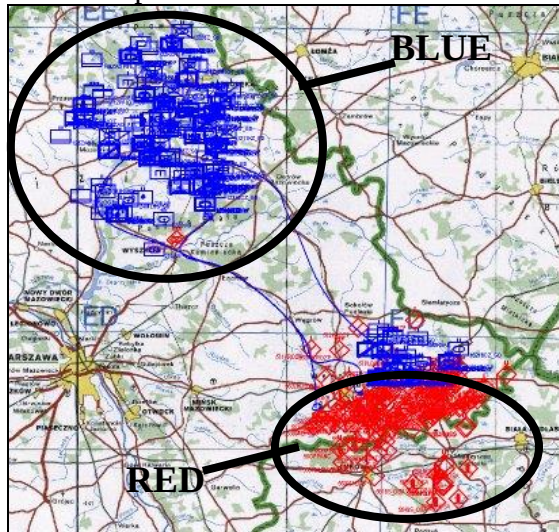


Figure 4. Initial tactical situation, 4:00am: two mechanized brigade of the “blue” side (121BZ and 123BZ) receive order to march

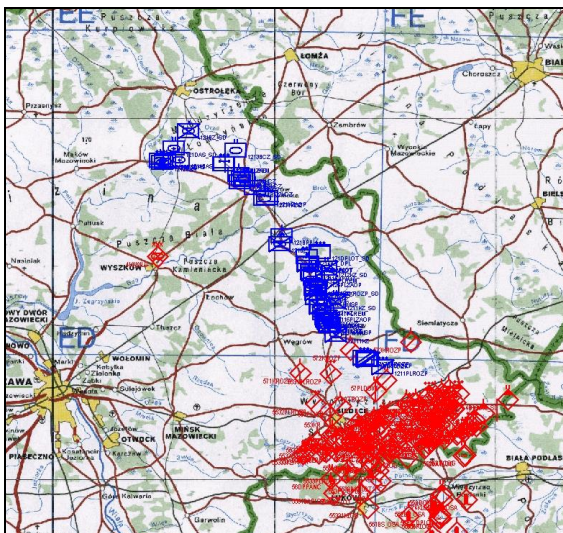


Figure 5. Location of the 121 BZ on the road, 5:50am

In the Figure 5 and Figure 6 location of 121BZ and 123BZ, respectively, after nearly 2 hours of march are presented.

Initial redeploying of the blue side is presented in the Figure 7. 121BZ is redeployed on the northern-east of the blue force redeploying area. On the south of 121 BZ is redeployed 123 BZ. In the Figure 8 location of 121BZ and

123BZ at 5:50am is shown.

In the Figure 9 percentage fuel level regarding to starting level (4825 liters) for selected unit (12111kz consisting 13 wheeled armored carrier “ROSOMAK”) during march on the distance 110 km, from 4:00 to 7:30 is presented. Fuel calculation during march simulation was done using formula (28).

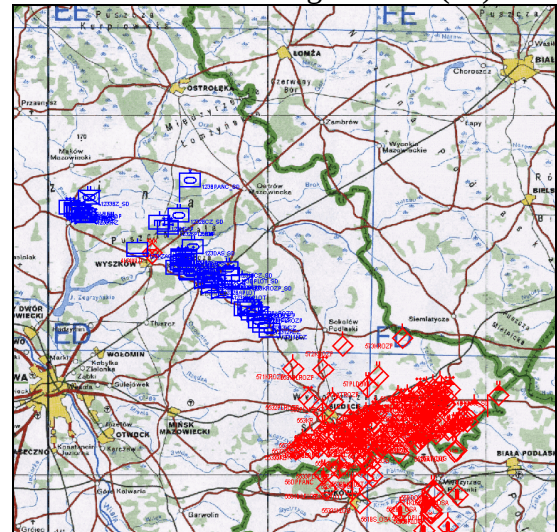


Figure 6. Location of the 123 BZ on the road, 5:50am

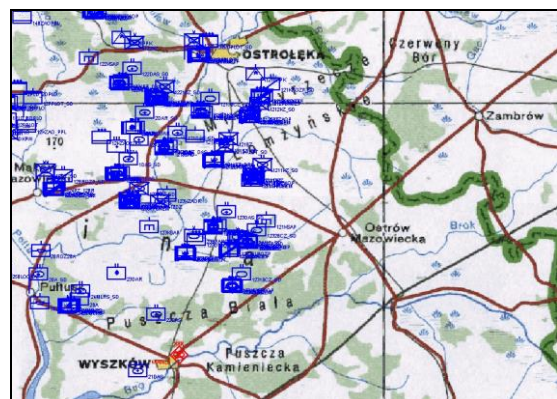


Figure 7. Initial redeploying of the blue side, 4:00am

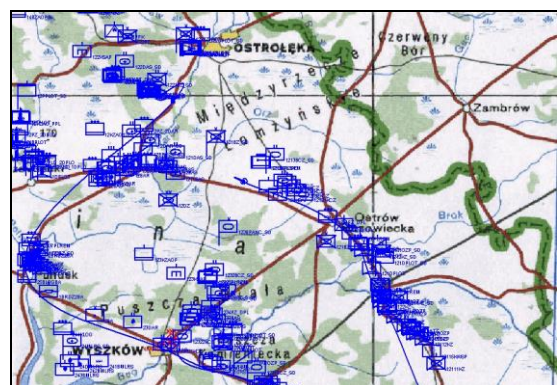


Figure 8. Location of the 121 BZ and 123 BZ, 5:50am

In the Table 3 average velocities between

selected march checkpoints for 121BZ and 123BZ are presented. Average march velocity is equal about 30km/h.

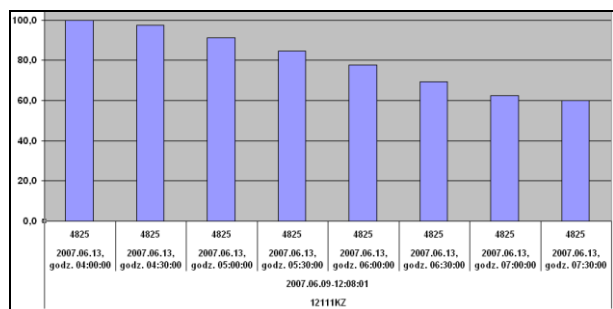


Figure 9. Percentage fuel level regarding to starting level (4825 liters) for 12111KZ (consisting of 13 wheeled armored carrier "ROSOMAK") during march on the distance 110 km, from 4:00 to 7:30

Table 3. Average velocities between selected march checkpoints for 121BZ and 123BZ (in km/h)

Unit	$S \Rightarrow P_S$	$P_S \Rightarrow P_D$	$P_D \Rightarrow D$	$S \Rightarrow D$
121BZ	12,32	39,65	18,24	29,54
123BZ	14,07	27,84	22,57	24,65

CONCLUSIONS

Presented methods being used in decision automata were implemented and were tested for different scenario (not only for march). In typical scenario number of squares in the network was about dozens of thousands. The models and methods described in the paper are being used in real simulation support system for military operational training on the corps-brigade level, which has been realized in Cybernetics Faculty of Military University of Technology in Warsaw for Polish Army [15]. Directions of development this part of the system may be as follows: effectiveness improvement of march planning algorithms [20], [21], extension of the movement planning models with models of location prediction and multicriteria analysis [18], improvement of movement synchronization many of units [19], [23].

ACKNOWLEDGEMENTS

This work was supported in part by the Military University of Technology in Warsaw under Grant PBW-571.

BIBLIOGRAPHY

- [1] Antkiewicz R., Najgebauer A., Tarapata Z., Rulka J., Kulas W., Pierzchała D., Wantoch-Rekowski R.: *The Automation of Combat Decision Processes in the Simulation Based Operational Training Support System*, Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA) 2007, ISBN 1-4244-0698-6, Honolulu (Hawaii, USA), April 1-5, 2007.
- [2] Antkiewicz R., Najgebauer A., Tarapata Z.: *The decision automata for command and control simulation on the tactical level*, Proceedings of The NATO Regional Conference on Military Communication and Information Systems, 07-10 October, Zegrze (Poland) 2003, (CD-ROM publication).
- [3] Benton J.R., Iyengar S.S., Deng W., Brener N., Subrahmanian V.S.: *Tactical route planning: new algorithms for decomposing the map*, Proceedings of the IEEE International Conference on Tools for AI, 6-8 November, Herndon 1995, pp. 268-277.
- [4] Brzeziński M.: *Logistics of Ground Forces*, Military University of Technology, Warsaw 1996 (in Polish).
- [5] Campbell C., Hull R., Root E., Jackson L.: *Route planning in CCTT*, in Proceedings of the 5th Conference on Computer Generated Forces and Behavioural Representation, Technical Report, Institute for Simulation and Training, pp. 233-244, 1995.
- [6] *Corps Battle Simulation (CBS), Version 1.6.0, Analyst's Guide, Volume 1, Ground*, U.S. Army Simulation, Training, and Instrumentation Command Orlando, Florida, June 2001.
- [7] Dockery J., Woodcock A.E.R.: *The Military Landscape, Mathematical Models of Combat*, Published by Woodhead Publishing Ltd., Cambridge, England, 1993
- [8] Hofmann H.W., Hofmann M.: *On the Development of Command&Control Modules for Combat Simulation Models on Battalion down to Single Item Level*. In: Proceedings of the NATO/RTO Information Systems Technology Panel (IST) Symposium "New

- Information Processing Techniques for Military Systems". NATO AC/329 (IST-017) TP/8, Istanbul, Turkey, 9-11 Oct. 2000, pp. 85-96.
- [9] James J., Sayrs B., Benton J., Subrahmanian V.S.: *Uncertainty management: keeping battlespace visualization honest*, <http://citeseer.nj.nec.com/386770.html>.
- [10] Karr C.R., Craft M.A., Cisneros J.E.: *Dynamic obstacle avoidance*, Proceedings of the Conference on Distributed Interactive Simulation Systems for Simulation and Training in the Aerospace Environment, The International Society for Optical Engineering, 19-20 April, Orlando (USA) 1995, pp. 195-219.
- [11] Kreitzberg T., Barragy T., Nevin B.: *Tactical movement analyzer: a battlefield mobility tool*, Proceedings of the 4th Joint Tactical Fusion Symposium, Laurel, 1990.
- [12] Lee J.J., Fishwick P.A.: *Simulation-based real-time decision making for route planning*, Proceedings of Winter Simulation Conference, 1995, 1087-1095.
- [13] Longtin M., Megherbi D.: *Concealed routes in ModSAF*, in Proceedings of the 5th Conference on Computer Generated Forces and Behavioural Representation, Technical Report, Institute for Simulation and Training, pp. 305-314, 1995.
- [14] Mohn H.: *Implementation Of A Tactical Mission Planner For Command And Control Of Computer Generated Forces In ModSAF*, <http://citeseer.ist.psu.edu/mohn94implementation.html>;
- [15] Najgebauer A.: *Simulation Based Operational Training Support System (SBOTSS) Zlocien*, Proceedings of the ITEC Conference, 20-24 April, London, 2004.
- [16] Petty M.D.: *Computer generated forces in Distributed Interactive Simulation*, Proceedings of the Conference on Distributed Interactive Simulation Systems for Simulation and Training in the Aerospace Environment, The International Society for Optical Engineering, 19-20 April, Orlando (USA) 1995, pp. 251-280.
- [17] Rajput S., Karr C.: *Unit Route Planning*, Technical Report IST-TR-94-42, Institute for Simulation and Training, Orlando (USA) 1994.
- [18] Tarapata Z.: *Selected multicriteria shortest path problems: an analysis of complexity, models and adaptation of standard algorithms*, International Journal of Applied Mathematics and Computer Science, Vol.17, No.2 (2007), 269-287.
- [19] Tarapata Z.: *Synchronization method of many objects movement in Computer Generated Forces systems*, Proceedings of the 7th NATO Regional Conference on Military Communication and Information Systems, 04-05 October, Zegrze (Poland) 2005, pp. 93-99.
- [20] Tarapata Z.: *Models and methods of movement planning and simulation in simulation aided system for operational training*, Proceedings of the 6th Regional Conference on Military Communication and Information Systems, 06-08 October, Zegrze (Poland) 2004, pp. 152-161.
- [21] Tarapata Z.: *Military route planning in battlefield simulation: effectiveness problems and potential solutions*, Journal of Telecommunications and Information Technology, N^o 4 (2003), pp. 47-56.
- [22] Tarapata Z.: *Modelling, optimisation and simulation of groups movement according to group pattern in multiresolution terrain-based grid network*, Proceedings of the 3rd NATO Regional Conference on Military Communication and Information Systems, 10-12 October, Zegrze (Poland) 2001, vol.I, 241-251.
- [23] Tarapata Z.: *Computer simulation of individual and grouped military objects redeployment*, Bulletin of Military University of Technology, N^o 1 (2000), pp. 147-162.
- [24] *The Analyst Guide, Joint Theater Level Simulation (JTLS)*, Version 1.65, Modern Aids to Planning Program (MAPP), Force Structure and Assessment Directorate (J-8), Joint Staff, The Pentagon, September 1988.

BIOGRAPHY

Zbigniew Tarapata (born in 1969) - graduated from Cybernetics Faculty (CF) at Military University of Technology (MUT) in Warsaw. He received his Master's degree in Computer Science in 1995 and Doctor's degree in the same field - in 1998. From 1995 to July 1999 he worked as an assistant, since July 1999 – as an assistant professor at Operations Research and Decision Support Division (OR&DSD) at CF of MUT. From October 2005 to December 2006 he was a Chief of OR&DSD. Since December 2006 he is a Vice-Director of Institute of Information Systems of CF of MUT. He is a member of scientific teams in CF on subjects: combat modelling and simulation, NEC/NCW, early warning identification of terrorist activities (NATO MSG-

026). His scientific interests and work are related to the following subjects: mathematical and simulation modeling of systems; methods and tools of operation research in combat and transportation (especially: combat modeling, simulation, optimization and prediction; transport optimization; graph and network optimization; transport algorithms complexity). See also: <http://tarapata.edu.pl>.